Performance Analysis of Two Frozen Image Based Backup/Restore Methods

Chung-Yen Chang
Network Appliance Inc.
Sunnyvale, CA 94089
chungyen@netapp.com

Yi-Chun Chu and Randy Taylor
VERITAS Software Corp.
Mountain View, CA 94043
{ycchu, randyt}@veritas.com

Abstract

Backup and restore are critical tasks performed on every system that holds important data. This paper evaluates the performance of two innovative backup methods based on frozen image technologies. VERITAS NetBackup Instant Recovery Option utilizes frozen images created from file system or volume manager utilities and manages these frozen images as backups. Because creating frozen images involve no actual copying of data, such backups are significantly faster. In addition, the frozen images reside on online storages, making restore from them much more efficient as well.

In this paper, we conducted backups and restores in a database environment to demonstrate the advantages of using the frozen image based backup/restore with the VERITAS File System's Storage Checkpoint and VERITAS Volume Manager's Volume Snapshot. With both methods, taking a full backup of a 26 gigabytes database took less than 4% of the time compared to traditional tape-based backup. The amount of time to restore different database objects from frozen images ranges from 3 to 47% of the time restoring from tapes.

While both backup and restore from frozen images are much more efficient than traditional backup methods, they are meant to complement, not to replace the traditional backups. This is because frozen images share common resources with the system they are protecting and are subjected to the same risks that might damage the data. The traditional backup method offers protection against a wider array of risks that can cause data loss and should be kept in as part of an overall data protection strategy.

1. Introduction

One of the major challenges in backup is the handling of open files during a hot backup (backups done while the application is updating the data) [1]. If a file is being actively written while a backup is performed, it is possible that portion of the file data has an inconsistent state in the backup image, compared to the rest of the file. This inconsistency may confuse the application and cause incorrect system behavior after a restore.

This data inconsistency problem is handled by open transaction manager [2, 3] or open file manager [4] features in the backup software. When a backup is taken, the open transaction manager or similar software monitors the updates and makes sure that the backup image contains a consistent view of the backed up files. This is typically done through creating a frozen image of the file system or the storage, which is a snapshot of the data at the time when the frozen image is created. The frozen image is then copied to the backup image such that changes in the live data set do not cause inconsistency in the backup image.

Frozen Images have been used by backup software as the source of the copy operation. As part of the industry trend toward leveraging the disk-based data protection [5-10], VERITAS NetBackup (NBU) 4.5 FP3 [2, 3] introduced a set of features that allow fast backup and restore directly using on-disk frozen images. The NBU Instant Recovery (IR) option [5] creates a frozen image of the live data through file system or volume manager utility and uses the frozen images directly as backup images. Because creating a frozen image does not involve actual copying of data, these operations are significantly faster than the traditional backup method, which moves substantial amount of data between storage media. The frozen images taken by the NBU IR features are kept online until they are expired or removed by a user. When performing a recovery from traditional backup method, data need to be copied back from a separate set of storage, which often requires time-consuming locating and mounting of the removable media. When performing a restore from online frozen images, the backup is readily accessible and allows a much quicker restore.

This paper studies two of the frozen image methods implemented in VERITAS NBU IR options: the VERITAS File System (VxFS) storage checkpoint and the VERITAS Volume Manager (VxVM) volume snapshot [11]. We compared the backup and restore performance of these two frozen image methods against the traditional tape-based

backups and restores. Our test results clearly indicate these IR features are very efficient in performing backups and restores.

Using frozen images as backups is not a replacement of the traditional backup methods. A VxFS storage checkpoint, which resides on the same physical media, does not survive from a media failure. A VxVM volume snapshot, which uses disks visible to the backup client, is also subject to many kinds of failures or disaster together with the systems they are intended to protect. The traditional tape-based backups offer better protection for data in those scenarios. Using frozen images as backups, however, has the advantage of very efficient backup and restore operations. They can be deployed at a higher frequency to complement the traditional backup methods as part of an overall data protection strategy.

The next section gives an overview of the operations involved in backup/restore with the frozen image studied in this paper. Section 3 analyzes the impact on overall system performance from the enabling technologies in VxFS and VxVM used in the two frozen image based backup methods. Section 4 describes the system and tests we ran. Section 5 presents the performance results from our backup/restore tests followed by the last section which concludes this study.

2. Operations of the Frozen Image Methods

This section briefly describes the operations involved in backups and restores using the two frozen image methods that we studied.

2.1 VxFS Storage Checkpoint

VERITAS File System (VxFS) provides four types of storage checkpoints for different applications. The type of storage checkpoint used for backup is the Data Storage checkpoint (referred as the storage checkpoint in the rest of this paper). The storage checkpoint is a point-in-time copy of a file system within the same file system. The creation of a storage checkpoint involves constructing the inode list and the block map that usually takes only a few seconds. After a storage checkpoint is created, if a data block in the primary file system is updated, the original data block is copied to the storage checkpoint before the primary file system is updated. This copy-on-write operation enables the storage checkpoint to maintain the exact view of the primary file system at the time the storage checkpoint was taken. The primary file system and the checkpoint share the same set of data blocks that have not been modified. The actual amount of storage required is thus proportional to the amount of data that has been updates since the checkpoint is created.

NBU uses the storage checkpoint as an alternative backup method. When a storage checkpoint based backup is initiated, NBU creates a storage checkpoint on the file system that contains the files to be backed up. Because creating a storage checkpoint is a very fast operation, the backup can be completed in a matter of seconds. When a restore is requested, NBU finds the corresponding storage checkpoint, mounts it, and copies the data required to perform the restore.

It should be noted that a storage checkpoint and the primary file system share the same media and can fail together if media failure occurs. Therefore, it is very important to use other data protection mechanisms such as the Redundant Array of Inexpensive Disks (RAID) to prevent data loss from media failure [11].

2.2 VxVM Volume Snapshot

VERITAS Volume Manager (VxVM) also provides a snapshot mechanism to create a point-in-time copy of a volume. To create a volume snapshot, a mirror volume (a storage device that is used to retain a copy of the primary data) is first created from the pool of free disk space and synchronized with the volume to be snapshot. This step is called *snapstart* in the VxVM terminology. Once the mirror volume is synchronized with the primary volume, subsequent data updates to the primary volume will also update the mirror volume to ensure data consistency. After the mirror and the original volumes are synchronized, a *snapshot* operation can be taken, which splits the mirror from the primary volume. All subsequent IO will change the primary volume only; hence leaving the split mirror with a copy of the primary volume at the time of the mirror split. The split mirror is called a snapshot of the primary volume and is used by NBU as another alternative to the tape-based backup.

Because a volume snapshot is a split-mirror, it consumes the same amount of storage space as the primary volume in the version of VxVM we tested[1]. When a backup is no longer needed, the mirror volume can be reused for future backups. The VxVM FlashSnap feature allows fast resynchronization between a volume and its snapshot, making the reuse of disk resource more efficient.

A restore from a volume snapshot is similar to in the storage checkpoint case. When a restore is requested, NBU

[1] In the latest version of VxVM (version 4.0), a new instant snapshot feature based on the copy-on-write technology is implemented. The instant snapshot feature only copies the data blocks into the snapshot area when they are first updated in the primary volume. This greatly reduces the storage overhead and eliminates the requirement for an initial synchronization before a snapshot can be taken.

finds the corresponding volume snapshot and temporarily mounts it so the data can be copied back.

3. Performance Impacts of Enabling the Frozen Images

The VxFS and VxVM frozen image technologies impact the system performance in ways different than the conventional backup methods. In a conventional backup, performance impact from the backup operation is visible only during the time data is copied. The VxFS storage checkpoint relies on a copy-on-write technology to maintain backups, which adds overhead to the file update operations. To use the VxVM volume snapshot for backups, an extra mirror has to be configured for the volume. Keeping the mirror in sync with the primary data volume also adds overhead to the write operations. The impacts on system performance associated with the overhead in both frozen image methods are analyzed here.

3.1 Performance Impacts of Using VxFS Storage Checkpoint

The analysis of performance impact for using VxFS storage checkpoints [12] as backups is straightforward. Update to a VxFS file system does not incur any performance overhead when no backup is taken. Once a storage checkpoint is created, subsequent writes to the backed up file system start to incur the copy-on-write overhead. The copy-on-write is only performed when a data block is updated for the first time after the snapshot is taken. Repeated updates to the same data block do not cause a copy-on-write operation. The magnitude of the overhead caused by a storage checkpoint depends on the workload and can vary from system to system. Workloads that contain a lot of updates tend to suffer more from this overhead. In previous studies, we observed a degradation of about 6% in a file server environment [13] and 20% in an OLTP environment with heavy writes in the I/O mix [14].

3.2 Performance Impacts of Using VxVM Volume Snapshot

To enable the VxVM volume snapshot feature [12], one needs to allocate sufficient disk space and creates an additional mirror on these extra disks. This extra mirror is used to hold the volume snapshot, which will serve as a backup after the mirror break off. During different stages of the operation, the extra mirror has different impacts on system performance.

A mirror used for a frozen image can be in one of three states at any given time: detached, attached, and synchronizing. When a mirror is in the detached state, it maintains a copy of the data at the time it was split from the data volume. A mirror is in this state when a backup is taken and kept. Before a backup is taken, a mirror needs to be in the attached state. When a mirror is attached, its content is always updated in parallel with the primary volume. Once a backup is expired, the corresponding mirror could be reused for future backups after it is synchronized with the primary volume again. When a mirror is in the synchronizing state, contents from the primary volume are actively copied to the mirror to bring the mirror and data volume to a consistent state.

Impacts on the system performance differ for the three mirror states. When a mirror is in a detached state (keeping a backup), its impact to the overall system performance is barely noticeable. When a mirror is in the attached state (anticipating a backup), all writes are duplicated to the mirror and cause an overhead to write traffic. The magnitude of this overhead depends on the workload and the physical layout of the volume. In a prior experiment [15], degradation to an Online Transactional Processing (OLTP) database workload was less than 1%. The biggest impact on the system performance is experienced when the mirror is synchronizing. A mirror has to be synchronized with the data volume before a backup can be taken. During the synchronization phase, data are copied from the primary volume to the snapshot mirror. The FastResync feature in VxVM greatly reduces the time required for re-synchronization but the IO performance is still degraded during the period when the copy is going on, due to contention on hardware resources. Our prior study showed this degradation for the snapback operation to be 1.4% in the same configuration we tested for an OLTP load [15].

4. System Configurations and Test Description

4.1 System Configuration

The following hardware and software are used in this study:

- Server – SUN E6500 with eight UltraSPAR-II processors (400 Mhz, 8 MB E-cache) and 8 GB physical memory.
- Disk Storage – Three SUN A5200 JBOD storage, each with 22 Seagate ST318304FC 18 GB 10,000 RPM disks.
- Tape Library - SUN L11000 tape library with 16 DLT 7000 tape drives.
- Software – Solaris 9 (release date August 2003), The 64-bit version of Oracle 9.2.0.4 is used. Key VERITAS software installed on the test system includes VERITAS Database Edition for Oracle 3.5, NetBackup 4.5 FP3, NetBackup for Oracle+,

NetBackup Core Frozen Images, and NetBackup Offhost and SAN Data Movement Services.

The E6500 served as the Master Server, Media Manager and backup client. It also ran the Oracle database. The three A5200 disk storages were connected to the host in a split bus mode via 6 SUN Socal Fibre Channel host bus adapters, with 11 disks on each controller. The L11000 tape library was connected to the host through 4 Crossroads 4200 SCSI-FC routers (4 drives on each) and a Broacade Silkworm 2800 FC switch. Two Qlogic 2200 HBA's were used on the host for the connection to the L11000 library (8 drives on each HBA).

Three volumes were configured on the A5200's to store the files for the test database. A 310 GB volume was used to store the datafiles. It was an 18-column stripe-mirrored volume on 36 disks. Each disk was mirrored in the volume to provide data redundancy for protection. To enable the volume snapshot operation, 18 additional disks were attached to this volume as a third mirror, bringing the total number of disks used for this volume to 54. Two 52 GB volumes hold the online and archive redo logs respectively. Both log volumes were 3-column stripe-mirrors built on 6 disks. The disks used to build the volumes were carefully selected so that no two volumes shared a disk and any two disks in the same mirror came from different controllers. This is to avoid unnecessary resource contention for writing to the mirrors.

4.2 Test Description

Simple backup and restore scenarios were set up for comparisons. In our tests, we performed backup and restore on a database that was built for running an Online Transactional Processing (OLTP) type of benchmark derived from the popular TPC-C benchmark. The test database had 9 tablespaces with a total of about 26 GB in size. We ran the backups through the Oracle Recovery Manager (RMAN) interface using three backup methods supported in NBU. After the backup was completed, we simulated three different system failure scenarios and restored the database from the backups.

The three test cases were simplified ways to simulate system failures of different magnitudes. In the first test, all data files in the database were removed from the system. In the second case, we removed all files related to one of the tablespaces (item), which had 20 files and an approximate size of 20 MB. In the third test, a single file (cust_0_0) was removed from the database. These tests simulate the situations when a customer has to restore the entire database, a tablespace, or a single datafile. The backup times for the different backup methods and the restore time involved in each of the three simulated failures were recorded for comparison.

A duplicate of the test database was stored on additional storage. Every time a test changed the state of the database (such as when a restore caused the storage checkpoint to contain data blocks generated from copy-on-write operations), we copied the test database back from the duplicate to ensure all tests were done in the same state, both from the database's and hardware's perspective. All tests were repeated eight times to validate the reproducibility of the tests. The data reported later are averages over all eight runs.

5. Test Results and Analysis

5.1 Backup Performance

Figure 1 shows the dramatic difference in taking a backup between the NBU IR features based on the frozen images and the traditional tape-based backup. Using frozen images as a backup, whether with a VxVM volume snapshot or VxFS storage checkpoint, backing up the 26 GB database took only a little over three minutes. Using tape as media, backing up the same database took almost one and a half hours, which is more than 26 times longer.

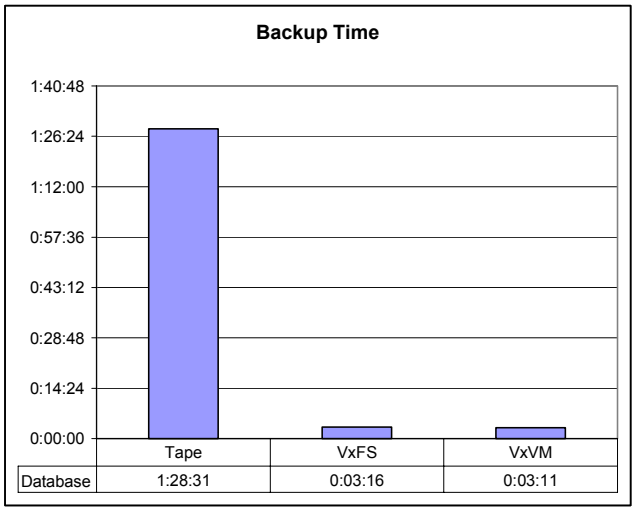

Figure 1 Comparison of time taken to backup the test database using three backup methods with Oracle RMAN

Table 1 highlights this comparison by showing the time taken to perform a backup with the two frozen image methods as a percentage of time it took with the tape-based backup. Both frozen image based backups took less than 4% of the time it took for the tape-based backup, a significant improvement.

Table 1 Backup time for NBU IR features as a percentage of tape-based backup method

VxFS Storage Checkpoint	VxVM Volume Snapshot
3.7%	3.6%

The reason for the much shorter backup time is the speed of creating a frozen image using a volume snapshot or storage checkpoint. While the traditional backup methods have to copy all the data to be protected to an image on the backup media, backup with frozen images is completed as soon as a frozen image is created. The frozen image only takes a few seconds to create; hence results in a much shorter backup time.

The actual saving in other environments could be even larger. For the tape-based backup, most of the 88 plus minutes were spent copying data from disks to tape. Because the two frozen image based backups were performed as Oracle RMAN proxy copy backups, Oracle has to prepare the database before backups can be taken. This preparation constitutes the majority of the 3+ minutes in the frozen image based backup cases. The actual operations of taking a volume snapshot or a storage checkpoint took only seconds. For environments in which this preparation is not required, such as a regular file system backup, the backup time could be even shorter.

5.2 Restore Performance

Time taken to restore different objects from the three backup methods was compared. Figure 2 illustrates the advantage of using the IR features when restoration of different object is concerned.

To restore the entire database (except the control file) from tape, it took close to 73 minutes on average. The same restore took an average of 34 minutes 45 seconds for the VxFS storage checkpoint based backup. Using VxVM volume snapshot as backup medium, the average restore time was 32 minutes and 21 seconds. Both frozen image cases were able to complete the restores in less than half of the time compared to tape-based operations as shown in Table 2 (47.6% for storage checkpoint and 44.3% for volume snapshot).

The advantage of the frozen image backups is more obvious when the object restored is small. This is true for both frozen images methods we tested. The largest difference occurs when the tablespace *item* was restored from the backup. Both IR features took only 3% of the time compared to the tape-based backup. When only one datafile was restored, the IR features took just 9.7% to complete the same operation versus restoring from tape.

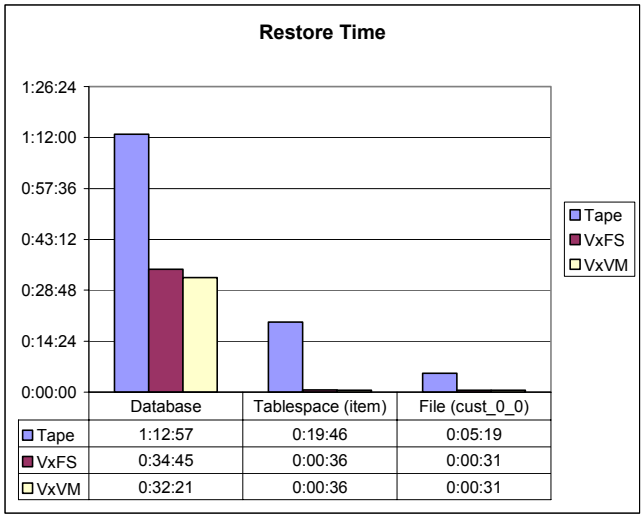

Figure 2 Comparison of times taken to restore different objects using three NBU features through Oracle RMAN

Table 2 Restore time for NBU IR features as a percentage of tape-based restores

Restored Object	Storage Checkpoint	Volume Snapshot
Database	47.6%	44.3%
Tablespace (item)	3%	3%
Datafile (cust_0_0)	9.7%	9.7%

Restoring from tape involves locating and mounting a tape, positioning the tape to the files to be restored, and actual copying of the data. All these operations contribute to the slower restore performance for the tape based restore. On-disk frozen images such as the two cases we tested here do not have to go through the media finding/mounting delay. Because they are on random access media, there is also no tape seek time involved before files can be copied.

The most obvious difference in our test cases is for restoring the tablesapce item. This tablespace consisted of 20 small files; each was 1 MB in size. While tape-based restores have to go through the tape mounting, and locating all 20 files on the tape, frozen image based restores can quickly copy all 20 files from the mounted frozen image. The fact that the frozen image is on disk media, which is faster than reading from tape, also helped the restore time.

The single file restore comparison used a file that is about 320 MB. In this case, seek time for finding the files on the tape was not as significant as the tablespace restore test above because there was only one file to be located. Nevertheless, the mounting of the tape is still unavoidable and the essential difference between copying from disks and tape still exists.

For restoring the entire database, the time difference between frozen image restores and tape-based restores became smaller but still significant. In this case, all files in the backup image were restored and therefore tape-based operations were not punished by the file seeking latency. The main contributing factor for the difference in this case is the speed of the media used: disks for IR, and tape in the other case.

6. Concluding Remarks

Two backup methods based on the frozen image technology were studied in this paper. We analyzed the impacts on system performance from the enabling technologies to create these frozen images and performed tests to assess their performance for the backup/restore operations. Our test results clearly indicate that backup/restore using frozen images are orders of magnitude faster and a viable complement to the traditional tape-based data protection method.

While the fast restore time is attractive, the very short backup time and low impact on system performance are the keys to allowing more frequent backups. Using these frozen image based backups, it is not necessary to push the backup operations to off-peak hours and finish them in a set time window.

However, these frozen image based backups do not offer the same level of data protection as the traditional backup methods because they are subjected to some of the risks that could potentially damage the frozen image along with the primary data that they are supposed to protect. As a best practice, these frozen image backups should be used along with traditional backup methods in an overall data protection strategy.

References

[1] A. L. Chervenak, V. Vellenki and Z. Kurmas, "Protecting File Systems: A Survey of Backup Techniques", Proc. of the Joint NASA and IEEE Mass Storage Conference, Mar. 1998.

[2] VERITAS NetBackup 5.1 System Administrator's Guide for Unix, Volume I, 268090, May 2004.

[3] VERITAS NetBackup 5.1 System Administrator's Guide for Unix, Volume II. 268091, May 2004.

[4] Open File Manager Version 9 User Manual, St. Bernard Software, Revision 9.4, Oct. 2004.

[5] VERITAS NetBackup 5.1 Advanced Client System Administrators Guide for UNIX and Windows, 268106, VERITAS Software Corporation, May 2004.

[6] VERITAS Software and Network Appliance Inc., "Using NetBackup Enterprise Server with NearStore", Network Appliance Inc. TR-3229, June, 2004.

[7] M. Fisch, "CLARiiON Disk Library Fits Right in to Backup", the Clipper Group, Apr. 2004.

[8] G. Crump, "Best Practices for Implementing Disk-to-Disk Backup", Computerworld, Feb. 2005.

[9] S. Hamilton, "Getting Disk into the Backup Process", Computer Technology Review, Jan. 2004.

[10] N. Wihelm-Olsen, J. Desai, G. Melvin, M. Federwisch, "Data Protection Strategies for Network Appliance Storage Systems", Network Appliance Inc., TR-3066, Apr. 2003.

[11] D. Patterson, G. Gibson, and R. Katz, "A Case for Redundant Arrays of Inexpensive Disks (RAID)", ACM SIGMOD 88, Chicago, June 1988, pp. 109-116.

[12] VERITAS FlashSnap Point-In-Time Copy Solutions Administrator's Guide 2.0, VERITAS Software Corporation, Feb. 2004.

[13] File System Checkpoint/Clone Performance Impact Study, VERITAS Engineering Report, July 2004

[14] VERITAS Database Edition 3.5 for Oracle Performance Brief – OLTP Comparison on Solaris 9 Platform, VERITAS Engineering Report, Nov. 2002.

[15] VERITAS Netbackup 4.5 Feature Pack 3 – Comparison of Three Backup Methods in an OLTP Environment, VERITAS Engineering Report, Nov. 2003.